Efficient and accurate computation of base generation allele frequencies

Aldridge, M. N., Vandenplas, J., & Calus, M. P. L.

INTERPRETIVE SUMMARY

Efficient and accurate computation of base generation allele frequencies

Aldridge

Several aspects of genomic prediction use allele frequencies. The current method is to calculate allele frequencies from the current genotyped population, however it is assumed they are equal to the allele frequencies in the pedigree base generation. We compared the current method, with best linear unbiased predictions and general least squares methods, to determine if there is a more accurate and equally efficient method, of calculating allele frequencies, that better represent the base generation. We concluded that the general least squares method using sparse relationship matrices should be adopted, as it is efficient, and more accurate than the current method.

Efficient and accurate computation of base generation allele frequencies

M.N. Aldridge,* J. Vandenplas,* and M.P.L. Calus*

*Wageningen University & Research, Animal Breeding and Genomics, 6700AH Wageningen, The Netherlands

Michael Nicholas Aldridge, Wageningen University & Research, Animal Breeding and Genomics, 6700AH Wageningen, The Netherlands, +31 643 835 587, michael.aldridge@wur.nl

ABSTRACT

Allele frequencies are used for several aspects of genomic prediction, with the assumption that these are equal to the allele frequency in the base generation of the pedigree. The current standard method, however, calculates allele frequencies from the current genotyped population. We compared the current standard method, with BLUP and general least squares (GLS) methods explicitly targeting the base population, to determine if there is a more accurate and still efficient method of calculating allele frequencies, that better represents the base generation. A dataset based on a typical dairy population was simulated for 325,266 animals, the last 100,078 animals in generations 9 to 12 of the population were genotyped, with 1,670 SNP markers. For the BLUP method, several SNP genotypes were analyzed with a multi-trait model by assuming a heritability of 0.99 and no genetic correlation among them. This method was limited by the time required for each BLUP to converge (approximately 6 minutes, per BLUP run of 15 SNPs). The GLS method had two implementations. The first implementation, using imputation on the fly and multiplication of sparse matrices, was very efficient, and required just 49 seconds and 1.3 GB of random access memory. The second implementation, using a dense full $\mathbf{A}_{22}^{-1}$ matrix, was very inefficient, and required more than one day wall clock time and over 118.2 GB of random access memory. When no selection was considered in the simulations, all methods predicted equally well. When selection was introduced, higher correlations between the estimated allele frequency and known base generation allele frequency were observed for BLUP (0.96 ± 0.01) and GLS (0.97 ± 0.01), compared to the current standard method (0.87 ± 0.01). The GLS method decreased in accuracy when introducing: incomplete pedigree with 25% of sires in the first five generations randomly replaced as unknown to erroneously identify founder animals (0.93 ± 0.01) and a further decrease for eight generations (0.91 ± 0.01). There was no change in accuracy when introducing 5% genotyping errors (0.97 ± 0.01), 5% missing genotypes (0.97 ± 0.01), or both 5% genotyping errors and missing genotypes (0.97 ± 0.01).

The GLS method provided the most accurate estimates of base generation allele frequency, and was only slightly slower compared to the current method. The efficient implementation of the GLS method, therefore, is very well suited for practical application and is recommended for implementation.

Key words: General least squares, best linear unbiased prediction, dairy cattle

INTRODUCTION

Allele frequencies are required for several processes in genomic prediction. The assumption for these processes is that the allele frequencies used is equal to the allele frequency of the base generation, commonly defined as the pedigree founders. For multi-step genomic evaluations, allele frequencies are used for the computation of model-based reliabilities of direct genomic values (VanRaden, 2008). However, VanRaden (2008) showed that there was limited impact on reliabilities of genomic predictions when using base generation or estimated allele frequencies. For single-step GBLUP, allele frequencies are used for the computation of genomic relationships (Aguilar et al., 2010, Christensen and Lund, 2010). The compatibility between pedigree and genomic relationships is an important issue in single-step GBLUP, as differences in the bases of both matrices may lead to bias of the predictions and reduce their accuracy. This possible bias can be overcome by making adjustments to the genomic relationships (Vitezica et al., 2011, Christensen, 2012, Gao et al., 2012). Using base generation allele frequencies to compute the genomic relationships is another possible approach towards getting pedigree and genomic relationships compatible. For estimating relationships among metafounders (pseudo-individuals used as founders in the pedigree, with an unknown sire and dam), the computation is based on the variance of the base generation allele frequencies

(Legarra et al., 2015), so estimating base generation allele frequencies accurately is essential for this process. However, it is standard practice to use allele frequencies calculated from the current genotyped population, because of the ease of computation.

An accurate and computationally efficient method of estimating base generation allele frequencies is desirable to replace the current standard method based on the currently genotyped population. Two methods have been proposed to explicitly estimate the base generation allele frequencies. The first method was to run, for each SNP, a best linear unbiased prediction (BLUP), where the heritability was close to 1 (e.g., 0.99 or smaller) (Gengler et al., 2007). The second method was, for each SNP, a general least squares estimator (GLS) using either sparse or dense matrices for the computation of the inverse of pedigree relationship sub-matrices (McPeek et al., 2004, Garcia-Baccino et al., 2017, Strandén et al., 2017). The BLUP and GLS methods are expected to be very similar because both use pedigree information, but we did not expect them to be exactly the same, although theoretically equivalent (e.g., Garcia-Baccino et al. 2017, Mrode 2005, Henderson, 1981), differences between estimates of BLUP and GLS methods could be due to the heritability different than one and the iterative solver used in the BLUP method. The objective of this study was to determine the most efficient and accurate method for estimating base generation allele frequencies when different scenarios likely to occur in real data are considered, including missing genotypes, genotyping errors and incomplete pedigree. We explored alternative implementations to improve the computational efficiency, with a multi-trait model for BLUP rather than the previously proposed single-trait model, such that these strategies could be applied with currently available and routinely used software.

MATERIALS AND METHOD

To achieve our objective datasets were simulated with a typical Holstein-like dairy population using QMSim (Sargolzaei and Schenkel, 2009). Each dataset was simulated with a historical population of 100,000 animals, decreasing to 500 animals over 2000 generations, and then rapidly increasing to 25,000 animals over 10 generations, this was to establish linkage disequilibrium in the base generation (average r^2 between adjacent markers = 0.41). The founder population and base generation for which the allele frequencies were to be estimated, consisted of 24,970 females and 30 sires, selected from the final historical generation. The population structure of the historical and founder population were selected to achieve an effective population size of ~100. The following 12 generations had a mutation rate of $2.5*10^{-5}$ (same mutation rate as the historical population), to ensure enough segregating markers in the final generations (Daetwyler et al., 2013), made random selections, random matings, and the same sex proportions in the founder population were maintained. The resulting pedigree included a total of 325,266 animals across 12 generations. This base simulation had no selection and was used as a control. Generations 9 to 12 were fully genotyped which included 100,078 animals. The genotyping included 1,670 SNPs with 250 QTL affecting the trait, with a uniform distribution of allele frequencies in the base generation, which were randomly positioned across 10 chromosomes, and each chromosome was 100cM in length. The number of markers was chosen to be similar to that in the additional simulations more likely to occur in reality.

Seven additional datasets were simulated using the same historical and founder population structure as the base simulation but with selection included, and depending on the scenario, errors or missing data were introduced to mimic reality (Table 1). All additional datasets used the base simulation, with selection for the last 12 generations based on high EBVs obtained

with BLUP and considering the true additive genetic variance, rather than randomly, and the 1,670 SNPs were positioned on a single chromosome of 100cM. The number of markers was selected for scaling to 1,670 SNPs on each of 30 chromosomes, to be representative of a commercial 50K chipset. In the datasets with selection, only a single chromosome was simulated to achieve a strong impact of selection on the change in allele frequency within a limited number of generations, illustrated by the allele frequency change between the base and the last genotyped generation (Figure 1). In the first dataset with selection it was assumed that the pedigree and all genotypes were known without error. The second dataset had an incomplete pedigree, created by randomly replacing 25% of sires in generations 1 to 5 as unknown parents, this was to replicate a situation in which pedigree records are lost and unknown sires are erroneously identified as base animals. The third dataset included extending the number of generations which randomly replaced 25% of sires as unknown parents up to generation 8. In the fourth dataset, genotyping errors were simulated, where a genotype is replaced by two randomly sampled alleles, at a rate of 5%. The fifth data simulation randomly introduced missing genotypes, at a rate of 5%. The sixth dataset included both the 5% erroneous genotypes and 5% missing genotypes. Finally a 50K SNP dataset was simulated with 30 chromosomes each with 1,670 SNPs randomly positioned.

In all scenarios a single dataset was simulated, where the results were evaluated across the 1,670 SNPs. So, the 1,670 SNPs served as replicates across which the results were evaluated. To evaluate if the dependency between SNPs may have affected the averaged results, we also selected a subset of SNPs including every 50th SNP and evaluated results for those separately. The average correlation between these 33 SNPs was 0.03 and were considered to be independent.

[INSERT TABLE 1 NEAR HERE]

[**INSERT FIGURE 1 NEAR HERE**]

However, it is standard practice to use allele frequencies calculated from the current genotyped population, because of the ease of computation.

The current standard method to calculate the allele frequencies of the genotyped population to be used as base generation allele frequencies was implemented with a Fortran program we developed, hereafter referred as "current method". The frequency of allele 1 of the i-th SNP, p_i, was computed as follows:

$$p_i = \frac{n_1}{2n}$$

Where n_1 is the number of occurrences for allele 1, and n is the total number of alleles. Another implementation was made where instead of using all genotyped animals, only the oldest genotyped generation was used, assuming they are a better representation of the base generation because they are closer connected to it.

The BLUP method involved evaluating the genotypes of each of the SNPs as a phenotype in a BLUP model, with the software MiXBLUP (Ten Napel et al., 2017). For each SNP the heritability was set to 0.99 following Gengler et al. (2007). To speed up the analyses, multiple SNPs were analyzed simultaneously by the means of a multi-trait model with zero genetic correlations among SNPs. To determine the optimum number of SNPs to be included in each run, a series of analyses were run with the number increasing from 1 to 60, in increments of 5. Based on the results of these analyses (Figure 2), the final BLUP analysis consisted of 111

BLUP runs of 15 SNPs, and one run of 5 SNPs, all performed in parallel. The MiXBLUP convergence criteria for the preconditioned conjugate gradient method was $1.0*10^{-12}$. The base generation allele frequency was estimated for each SNP as $\hat{\mu}/2$, where $\hat{\mu}$ was the estimate of the general mean of the model. Simulated missing genotypes were considered as missing phenotypes in the analysis.

The GLS equivalent uses the method proposed by McPeek et al. (2004) and implemented by Strandén et al. (2017) and Garcia-Baccino et al. (2017). Whereby for the i-th SNP: $\hat{\mu}_i = \left(\mathbf{1}'\mathbf{A}_{22}^{-1}\mathbf{1}\right)^{-1}\mathbf{1}'\mathbf{A}_{22}^{-1}\mathbf{z}_i$,

where $\mathbf{1}$ is a vector of ones, $\mathbf{A}_{22}^{-1}$ is the inverse pedigree relationship matrix of genotyped animals and $\mathbf{z}_i$ is a vector of genotypes coded as 0, 1, and 2. Two implementations of this method were made. Our first implementation referred to as "GLS_Sparse", was similar to that of Strandén et al. (2017), in the sense that the vector $\mathbf{t} = \mathbf{A}_{22}^{-1}\mathbf{1}$ was first computed as a multiplication of sparse matrices by the vector $\mathbf{1}$, followed by the trivial computation of the scalar $\alpha = (\mathbf{1}'\mathbf{A}_{22}^{-1}\mathbf{1})^{-1} = (\mathbf{1}'\mathbf{t})^{-1}$, and by the multiplication of a matrix and vector, that is $\hat{\mu} = \alpha\mathbf{t}'\mathbf{Z}$. The vector $\mathbf{t}$ can be efficiently computed as follows Strandén et al. (2017):

$$\mathbf{t} = \mathbf{A}_{22}^{-1}\mathbf{1} = \left[\left[\mathbf{A}^{22}\mathbf{1}\right] - \left[\mathbf{A}^{21}\left[\left(\mathbf{A}^{11}\right)^{-1}\left[\mathbf{A}^{12}\mathbf{1}\right]\right]\right]\right]$$

where, $\mathbf{A}^{ij}$ are submatrices of $\mathbf{A}^{-1}$, a value for *i* and *j* of 1 denotes non-genotyped animals while a value of 2 denotes genotyped animals, and the brackets [...] indicate the order of the matrix-vector operations.

In our implementation, MKL subroutines were used for the matrix-vector multiplications, and Intel MKL-PARDISO (Schenk et al., 2001) was used to compute $\mathbf{x} = (\mathbf{A}^{11})^{-1}\left[\mathbf{A}^{12}\mathbf{1}\right] = (\mathbf{A}^{11})^{-1}\mathbf{v}$ by solving $\mathbf{A}^{11}\mathbf{x} = \mathbf{v}$. In GLS_Sparse, missing genotypes were replaced with the current genotype mean, computed across all animals with observed genotype for this locus. The second implementation of the GLS method, instead calculates the full $\mathbf{A}_{22}^{-1}$ directly using Calc_grm (Calus and Vandenplas, 2016), hereto referred as "GLS_Full". This approach may mimic an approach where a user would use available software.

All computations were run on a high performance cluster (HPC). The HPC was designed with 48 nodes: 16 cores, 64 GB memory, Intel Xeon, and 2.2 GHz. A single thread was used for the current, BLUP, and GLS_Sparse methods. For the computation of $\mathbf{A}_{22}^{-1}$ with Calc_grm, one node with 64 cores, 1 TB memory, AMD Opteron, and 2.3 GHz was used. A total of 16 threads were used for Calc_grm, but the implementation of the full $\mathbf{A}_{22}^{-1}$ in $\hat{\mu}_i = \left(\mathbf{1}'\mathbf{A}_{22}^{-1}\mathbf{1}\right)^{-1}\mathbf{1}'\mathbf{A}_{22}^{-1}\mathbf{z}_i$ was done with a single thread on the same nodes as the other methods.

To determine if one of the methods of estimating base generation allele frequency should be used to replace the current method, it needs to be efficient and at least as accurate. To determine efficiency, both the wall clock time and total processing time were compared between methods. Wall clock time varied depending on the number of CPUs used, if parallel processing is used, and if the process had been optimized. That is why it was also important to consider the processing time, which accounts for the total time used across all CPUs and processes. Similarly for computational efficiency, the total Random Access Memory (RAM) used was also reported to compare memory requirements. Wall clock time, total processing time, and total RAM were

recorded as the maximum job requirements, as reported by the HPC. Accuracy was determined by the correlation of the known base generation allele frequency from QMSim, and the estimated allele frequency.

RESULTS

The results for efficiency are only presented for the base simulation without selection as the results were similar for the other simulations (Table 2). We observed the current method of estimating base generation allele frequency using all genotyped animals is fast (3 seconds). Using the same method but with only animals from the oldest genotyped generation was even faster (1 second). Using the GLS method with GLS_Sparse required more time but we still considered it to be efficient (49 seconds). Using methods BLUP (35 minutes), or the full $\mathbf{A}_{22}^{-1}$ with GLS_Full (over 1 day), were not efficient compared to the current method. Finally, the GLS_Sparse method was also tested with the 50K SNP dataset which required 6 minutes of processing time.

[**INSERT TABLE 2 NEAR HERE**]

The processing time for the current method, and only the oldest genotyped generation, had no additional time requirements compared to the wall clock time. The GLS_Sparse method was the fastest alternative method (49 seconds). The total processing time for the BLUP analysis (12 hours, 42 minutes), was an accumulative amount of time, caused by the total number of individual runs required in MiXBLUP of 15 correlated SNPs (minimum time per run <5 minutes). Less than 10 seconds per SNP was required for MiXBLUP runs with between 5 and

20 SNPs. For a run with 60 SNPs, approximately 15 seconds per SNP was required during solving (Figure 2). The total processing time was increased for 60 SNPs (13 hours, 42 minutes) due to the minimum time per run (approximately 30 minutes), but there was no significant difference in memory requirements. The total processing time for the GLS method using the full $\mathbf{A}_{22}^{-1}$ was exceptionally demanding (over 19 days), the majority of which was used to invert the $\mathbf{A}_{22}$ matrix using Calc_grm.

[INSERT FIGURE 2 NEAR HERE]

The total RAM required for each method was closely related to the total processing time (Table 2). The current method required very little memory (<0.1 GB), and only using the oldest genotyped generation, even less (<0.1 GB). GLS_Sparse required more RAM (1.3 GB) but was still computationally efficient. When the 50K SNP dataset was used, GLS_Sparse required up to 37.6 GB. The RAM requirements for the BLUP analysis with 1,670 SNPs was large (49.0 GB) due to the individual BLUP runs of 15 SNPs which required 0.4 GB each. Using the full $\mathbf{A}_{22}^{-1}$ for the GLS validation was the most demanding (118.2 GB), again primarily due to storing the full $\mathbf{A}_{22}^{-1}$ matrix and its inverse with Calc_grm (78.4 GB).

For all datasets and methods, there was no significant difference in accuracy between the full 1,670 SNPs and the subsets of 33 independent SNPs, therefore, only the results for the full datasets are presented. When using the base simulation with no selection, the accuracies, computed as correlations between the estimated allele frequency and the known simulated frequency, were not different to one (0.99 ± 0.01), for all methods (Table 3). Significant

differences in accuracy between methods were observed for simulations that included selection. When using the current method with all genotyped animals the accuracy decreased to 0.87 ± 0.01, by only using the oldest genotyped generation, the accuracy was slightly increased but was not significantly different (0.88 ± 0.01). We observed that both the BLUP (0.94 to 0.97) and GLS (0.93 to 0.97) methods significantly increased the accuracy for all data simulations under selection. There was no significant difference between the BLUP and GLS methods with a correlation of 0.99 ± 0.01 observed with the base simulation under selection (Figure 3). For both the BLUP and GLS method, the estimated allele frequencies were more similar to the true base generation allele frequency for allele frequencies <0.10 and >0.90, while larger differences were observed where the true allele frequency was closer to 0.50 (Figure 4).

[INSERT TABLE 3 NEAR HERE]

[INSERT FIGURE 3 NEAR HERE]

[INSERT FIGURE 4 NEAR HERE]

When founders are erroneously identified in the pedigree between generations 1 and 5 the accuracy is still improved with both BLUP (0.94 ± 0.01) and GLS (0.93 ± 0.01) compared to the current method (0.87 ± 0.01). When the incomplete pedigree is continued up to generation 8, the accuracy was decreased for the BLUP and GLS methods (0.91 ± 0.01). The accuracy with the incomplete pedigree was lower compared to the other data simulations. Introducing 5% missing genotypes or 5% genotyping errors did not affect the accuracy (0.97 ± 0.01). When both the 5% missing and 5% genotyping errors were included none of the methods were affected. The missing genotype rate was reanalyzed for the GLS_Sparse method to see what effect different missing genotype rates (between 1 to 10%) had on the accuracy of estimation

(Figure 5). The GLS_Sparse method was very robust, even up to 10% missing genotypes the accuracy was not significantly different to 0.97, although the accuracy did start to decrease after 8% missing genotypes (0.96).

[INSERT FIGURE 5 NEAR HERE]

DISCUSSION

The objective of this study was to compare methods for estimating base generation allele frequencies in terms of efficiency and accuracy. The only method both efficient and accurate was the GLS method using GLS_Sparse. With wall-clock and processing times less than one minute, it can be implemented in routine genomic evaluations without jeopardizing overall efficiency. The RAM requirements for the GLS_Sparse are linearly related to the number of SNPs, as shown by the results obtained with the 50K SNP dataset. While the time requirement is already limited (<10 minutes for the 50K dataset), it could be even further improved by using parallel processing, since the MKL library and PARDISO are multi-threaded. For example, the wall clock time was reduced to <5 minutes when using 4 threads. The 50K SNP was not analyzed with the BLUP method but would require 3,340 runs of 15 SNPs each. Assuming each run was equal to the mean time required (0-00:06:20), the required processing time would be over 14 days, and the observed wall clock time would be limited by the number of parallel MiXBLUP runs that can be run at the same time. As already demonstrated the GLS_Full was already inefficient for 1,670 SNPs and no attempt to analyze the 50K SNP dataset with GLS_Full was made nor is it recommended. It is worth noting that computing explicitly $\mathbf{A}_{22}^{-1}$ is not strictly necessary for GLS_Full, because we need the product $\mathbf{t} = \mathbf{A}_{22}^{-1}\mathbf{1}$, which can be

more time-efficiently computed as $\mathbf{t} = \mathbf{L}^{-1'}[\mathbf{L}^{-1}\mathbf{1}]$, with the matrix $\mathbf{L}$ being the Cholesky factor of $\mathbf{A}_{22}$. This strategy would request the same amount of memory as GLS_Full, and will be considerably faster than GLS_Full. Even then it would still be computationally much less efficient than GLS_Sparse.

Importantly, GLS_Sparse is more accurate than the current method that simply computes the allele frequency in the current genotype data. It is recommended that the GLS_Sparse method is implemented, when using allele frequencies for genomic prediction processes, where the assumption requires base generation frequencies. Arguably, with increasing amounts of genotype data available, the estimated base generation allele frequencies will not change as much over time as the allele frequencies in the genotype data. In practical implementations, one could consider not to re-estimate base generation allele frequencies for every run of the genetic evaluation. Instead, they could be re-estimated for instance every time that the variance components of the model are re-estimated. Any possible fluctuations in results (as an example, genomic estimated breeding values), caused by changing allele frequencies when new genotyped animals are added and when using the current method, would therefore only occur when the frequencies are re-estimated and not for every evaluation.

There was no significant difference in accuracy between the GLS and BLUP methods, as both use the pedigree information. GLS and BLUP had high correlations with the known base generation allele frequencies, estimates are virtually the same with incomplete pedigree, but the estimates from the two methods were different with both genotyping errors and missing genotype datasets. Additional analyses with BLUP (results not shown), mimicking the GLS implementation by using a heritability of 0.99999 and replacing missing genotypes by the

average genotype in the data, confirmed that the difference between GLS and BLUP is due to using a non-unity heritability in BLUP, and by replacing genotypes in GLS with the average (which is probably worse than putting it to missing in BLUP). However in many practical applications replacing missing values in the GLS method will probably be unnecessary as imputation is common practice. When a considerable number of genotyping errors is present, the BLUP method may be better able to deal with this, as it has been suggested to be robust against genotyping errors (Gengler et al., 2007). In such cases the heritability used should probably reflect the proportion of genotyping errors, and a value lower than our value of 0.99 may be more appropriate. In fact, the heritability of the genotypes of each SNP could be estimated to assess its quality in the first place (Forneris et al., 2015).

Results for the simulated scenario with selection did indicate that estimated allele frequencies deviated considerably, up to 0.25 units, from the actual values. Observed deviations were larger for allele frequencies closer to 0.50 and limited at <0.10 or >0.90. This is because the estimates of allele frequencies closer to 0 or 1 were on one side bounded to stay within the parameter space. The simulations employed were rather extreme in the sense that changes in allele frequencies up to 0.7 units were observed across 12 generations of selection. In real-life breeding programs it is unlikely to see so many loci with such big changes in allele frequencies in such a short time frame, so the expected deviations of the estimated from the true base allele frequencies are expected to be smaller.

The only partial limitation observed with GLS_Sparse method, was that SNPs that had a minor allele frequency below 0.001 in both the base generation and current population, would sometimes result in an estimate outside of the parameter space. This was also observed with the

BLUP method. There were three SNPs in the base simulation without selection, that were outside of the parameter space (outside parameter space by <0.001). Similar numbers of SNPs were observed outside the parameter space for the other data simulations. These SNPs also had known minor allele frequencies in the base generation of <0.001 and known minor allele frequencies in the generation 12 of <0.001. Such estimates have been observed by VanRaden (2008) and Makgahlela et al. (2013), which suggested that these outliers are due to the use of linear algebra, instead of nonlinear probabilities.

Alternatively the current method was used to filter SNPs with a minor allele frequency (<0.01) before running GLS_Sparse. The only benefit was it did remove the SNPs with estimates that previously were outside the parameter space (results not shown). Realistically those SNPs would be removed during standard processing practices before being used in GLS_Sparse, and estimates outside the parameter space are not expected to occur. If the base allele frequency is needed for all markers, it may be necessary to assume they are fixed by assigning missing or zero to markers outside the parameter space. We conclude that the GLS_Sparse method is efficient, robust and accurate within the range of allele frequencies 0.01 to 0.99.

When animals were erroneously identified as founders due to incomplete pedigree we observed a significant decrease in accuracy for the BLUP and GLS methods. The accuracy was decreased further when removing the pedigree for 25% of the animals up until generation 8, which was the last non-genotyped generation. This effectively meant that animals from later generations were added to the base, and because allele frequencies changed across generations, the estimates represented some sort of average across generations instead of those in the base generation. It is important to note that the accuracy for the BLUP and GLS methods were still

greater compared to the current method. The accuracy for such cases could be improved by taking into account the different base populations, by implementing the GLS_Sparse method with genetic groups. This could be done by replacing the vector **1** in the different formula by a matrix **Q** that contains the expected fraction of each genetic group for each genotyped individual, that is $\hat{\boldsymbol{\mu}}_i = \left(\mathbf{Q}'\mathbf{A}_{22}^{-1}\mathbf{Q}\right)^{-1}\mathbf{Q}'\mathbf{A}_{22}^{-1}\mathbf{z}_i$ with $\hat{\boldsymbol{\mu}}_i$ being a vector of estimates of base allele frequencies for all genetic groups (Gengler et al., 2007, VanRaden, 2008, Makgahlela et al., 2013, Garcia-Baccino et al., 2017). The strategies used for GLS_Sparse are readily extendable for the computation of $\mathbf{Q}'\mathbf{A}_{22}^{-1}$ and $\left(\mathbf{Q}'\mathbf{A}_{22}^{-1}\mathbf{Q}\right)^{-1}$.

Allele frequencies are required for several processes in genomic prediction. This includes computation of model-based reliabilities of direct genomic values in the context of multi-step genomic evaluations, computation of genomic relationships to be used in single-step GBLUP, computation of relationships among metafounders, and compatibility between the pedigree and genomic relationship matrices. The bias due to compatibility between the relationship matrices can be overcome by adjusting the genomic relationship by blending with the pedigree matrix (Gao et al., 2012), or shifting the genomic relationships by an analytically derived constant (Vitezica et al., 2011). Alternatively the pedigree relationship matrix can be adjusted by scaling it to the genomic relationship matrix (Christensen, 2012). While these adjustments for the relationship matrices could be more efficient than the computation of base allele frequencies when performing a genomic evaluation, it can be assumed that the computation of base allele frequencies could be performed only once for multiple successive genomic evaluations (e.g., at the same rate as variance components estimation), which would reduce its costs even further.

CONCLUSIONS

There are a number of benefits for calculating base generation allele frequencies using the general least squares method, with a pedigree relationship matrix computed using sparse matrices. It is fast, so that practical application is appropriate and would not delay other processes. It is accurate in estimating base generation allele frequencies under a number of different scenarios, thereby better fulfilling the assumptions of genomic prediction processes than the current method. We recommend base generation allele frequencies be estimated using a GLS method implemented with sparse matrices for $\mathbf{A}_{22}^{-1}$, and replacing any missing genotypes with the mean allele frequency calculated from the genotyped population, or with imputed values.

ACKNOWLEDGEMENTS

This study was financially supported by the Dutch Ministry of Economic Affairs (TKI Agri & Food project 16022) and the Breed4Food partners Cobb Europe, CRV, Hendrix Genetics and Topigs Norsvin. The use of the HPC cluster has been made possible by CAT-AgroFood (Shared Research Facilities Wageningen UR).

REFERENCES

Aguilar, I., I. Misztal, D. Johnson, A. Legarra, S. Tsuruta, and T. Lawlor. 2010. Hot topic: a unified approach to utilize phenotypic, full pedigree, and genomic information for genetic evaluation of Holstein final score. J. Dairy Sci. 93:743-752. https://doi.org/10.3168/jds.2009-2730

Calus, M. and J. Vandenplas. 2016. Calc_grm—a program to compute pedigree, genomic, and combined relationship matrices. ABGC, Wageningen UR Livestock Research.

Christensen, O. F. 2012. Compatibility of pedigree-based and marker-based relationship matrices for single-step genetic evaluation. Genet. Sel. Evol. 44:37. https://doi.org/10.1186/1297-9686-44-37

Christensen, O. F. and M. S. Lund. 2010. Genomic prediction when some animals are not genotyped. Genet. Sel. Evol. 42:2. https://doi.org/10.1186/1297-9686-42-2

Daetwyler, H. D., M. P. Calus, R. Pong-Wong, G. de los Campos, and J. M. Hickey. 2013. Genomic prediction in animals and plants: simulation of data, validation, reporting, and benchmarking. Genetics 193:347-365. https://doi.org/10.1534/genetics.112.147983

Forneris, N. S., A. Legarra, Z. G. Vitezica, S. Tsuruta, I. Aguilar, I. Misztal, and R. J. Cantet. 2015. Quality control of genotypes using heritability estimates of gene content at the marker. Genetics 199:675-681. https://doi.org/10.1534/genetics.114.173559

Gao, H., O. F. Christensen, P. Madsen, U. S. Nielsen, Y. Zhang, M. S. Lund, and G. Su. 2012. Comparison on genomic predictions using three GBLUP methods and two single-step blending methods in the Nordic Holstein population. Genet. Sel. Evol. 44:8. https://doi.org/10.1186/1297-9686-44-8

Garcia-Baccino, C. A., A. Legarra, O. F. Christensen, I. Misztal, I. Pocrnic, Z. G. Vitezica, and R. J. Cantet. 2017. Metafounders are related to F st fixation indices and reduce bias in single-step genomic evaluations. Genet. Sel. Evol. 49:34. https://doi.org/10.1186/s12711-017-0309-2

Gengler, N., P. Mayeres, and M. Szydlowski. 2007. A simple method to approximate gene content in large pedigree populations: application to the myostatin gene in dual-purpose Belgian Blue cattle. Animal 1:21-28. https://doi.org/10.1017/S1751731107392628

444 Legarra, A., O. F. Christensen, Z. G. Vitezica, I. Aguilar, and I. Misztal. 2015. Ancestral
445 relationships using metafounders: finite ancestral populations and across population
446 relationships. Genetics 200:455-468. https://doi.org/10.1534/genetics.115.177014
447 Makgahlela, M., I. Strandén, U. Nielsen, M. Sillanpää, and E. Mäntysaari. 2013. The
448 estimation of genomic relationships using breedwise allele frequencies among animals
449 in multibreed populations. J. Dairy Sci. 96:5364-5375.
450 https://doi.org/10.3168/jds.2012-6523
451 McPeek, M. S., X. Wu, and C. Ober. 2004. Best linear unbiased allele-frequency estimation in
452 complex pedigrees. Biometrics 60:359-367. https://doi.org/10.1111/j.0006-
453 341X.2004.00180.x
454 Sargolzaei, M. and F. S. Schenkel. 2009. QMSim: a large-scale genome simulator for
455 livestock. 25:680-681. https://doi.org/10.1093/bioinformatics/btp045
456 Schenk, O., K. Gärtner, W. Fichtner, and A. Stricker. 2001. PARDISO: a high-performance
457 serial and parallel sparse linear solver in semiconductor device simulation. Future
458 Gener. Comp. Sy. 18:69-78. https://doi.org/10.1016/S0167-739X(00)00076-5
459 Strandén, I., K. Matilainen, G. Aamand, and E. Mäntysaari. 2017. Solving efficiently large
460 single-step genomic best linear unbiased prediction models. J. Anim. Breed. Genet.
461 134:264-274. http://dx.doi.org/10.1111/jbg.12257
462 Ten Napel, J., J. Vandenplas, M. Lidauer, I. Stranden, M. Taskinen, V. Mäntysaari, M. P. L.
463 Calus, and R. F. Veerkamp. 2017. MiXBLUP, user-friendly software for large genetic
464 evaluation systems – Manual Wageningen, the Netherlands V2.1-2017-08.
465 http://www.mixblup.eu/documents/Manual%20MiXBLUP%202.1_June%202017_V2.
466 pdf
467 VanRaden, P. M. 2008. Efficient methods to compute genomic predictions. J. Dairy Sci.
468 91:4414-4423. https://doi.org/10.3168/jds.2007-0980

469 Vitezica, Z., I. Aguilar, I. Misztal, and A. Legarra. 2011. Bias in genomic predictions for
470 populations under selection. Genet. Sel. Evol. 93:357-366.
471 https://doi.org/10.1017/S001667231100022X

472

473

TABLES

Table 1: Summary of the structure and errors for the different data simulations.

Dataset	Chromosomes	Selection	Data error
Base simulation	10	No selection	No errors
Base simulation	1	High EBVs	No errors
Incomplete pedigree	1	High EBVs	25% of sires in generation 1 to 5 are randomly replaced as unknown
Incomplete pedigree	1	High EBVs	25% of sires in generation 1 to 8 are randomly replaced as unknown
Genotyping errors	1	High EBVs	5% of genotypes are replaced by randomly sampled alleles
Missing genotypes	1	High EBVs	5% of genotypes are randomly replaced as missing
Errors and missing	1	High EBVs	Both 5% genotyping errors and missing genotypes
50K SNPs	30	No selection	No errors

Table 2: Computational time (day-hour:minute:second) and memory requirements to complete the full process of each method for the base simulation without selection.

Method	Process time	Wall clock time	Random Access Memory
Current method	0-00:00:03	0-00:00:03	<0.1 GB
Oldest genotyped animals	0-00:00:01	0-00:00:01	<0.1 GB
111 MiXBLUPs[1]	0-12:42:47	0-00:10:50	48.9 GB
1 MiXBLUP of 15 SNPs[2]	0-00:06:20	0-00:06:20	0.4 GB
GLS_Sparse	0-00:00:49	0-00:00:49	1.3 GB
GLS_Full	19-23:05:09	1-08:25:24	118.2 GB

[1]Requirements for 111 BLUP runs including 15 SNPs and 1 run including 5 SNPs.

[2]Average requirements for 111 BLUP runs, including 15 SNPs.

483 Table 3: Correlation between the known base generation allele frequency and estimated allele

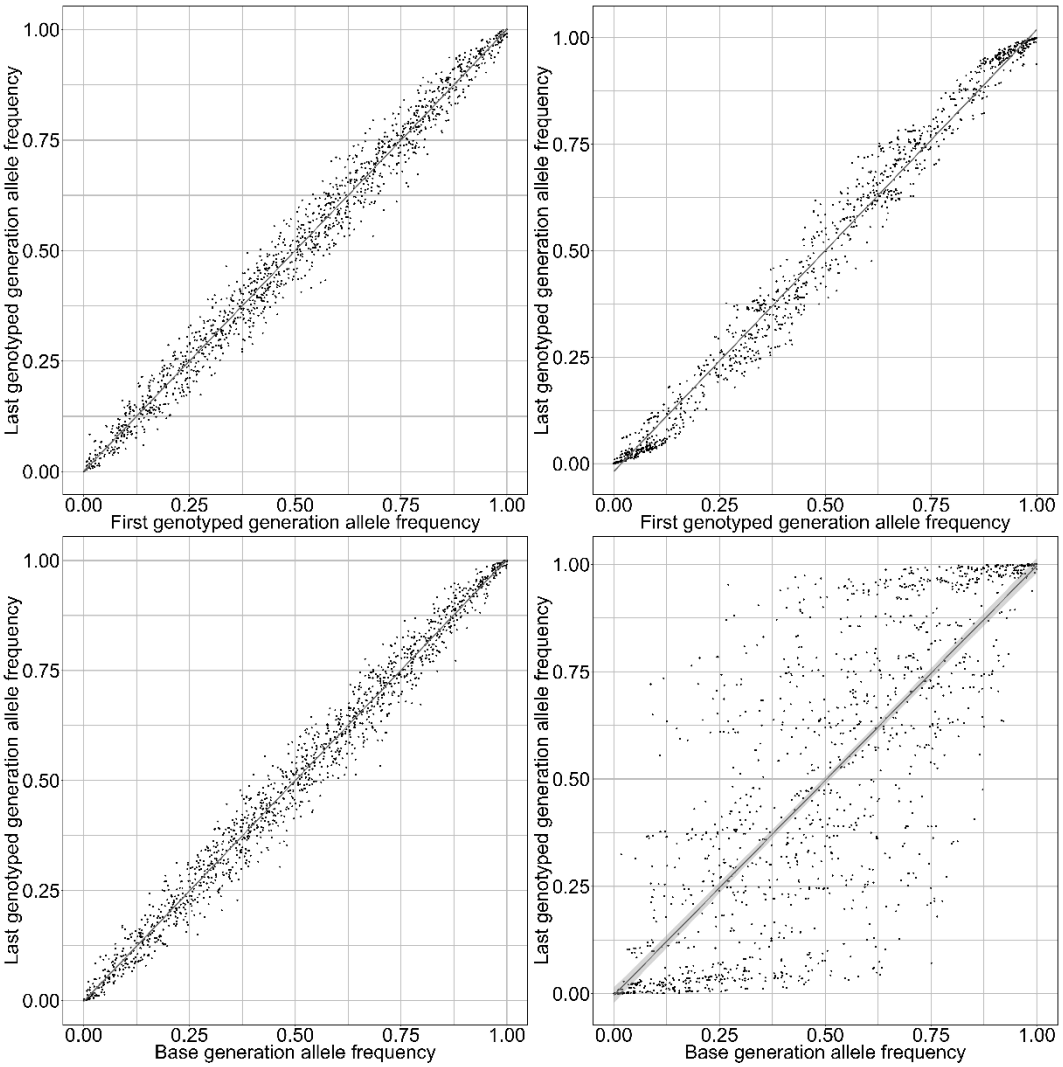
484 frequency, all standard errors were < 0.01.

Method	No selection	With selection					
	Base simulation	Base simulation	Incomplete pedigree		Genotype errors	Missing genotypes	Errors and missing
			Generations 1 to 5	Generations 1 to 8			
Current method	0.99	0.87	0.87	0.87	0.87	0.87	0.87
Oldest generation genotyped	0.99	0.88	0.88	0.88	0.88	0.88	0.88
MiXBLUP	0.99	0.96	0.94	0.91	0.97	0.97	0.97
GLS_Sparse	0.99	0.97	0.93	0.91	0.97	0.97	0.97
GLS_Full	0.99	0.97	0.94	0.91	0.97	0.97	0.97

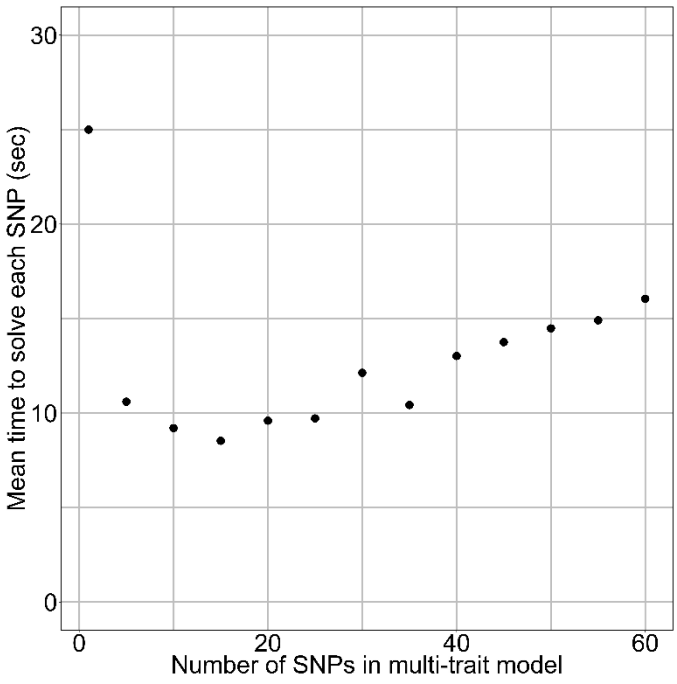
485

FIGURES

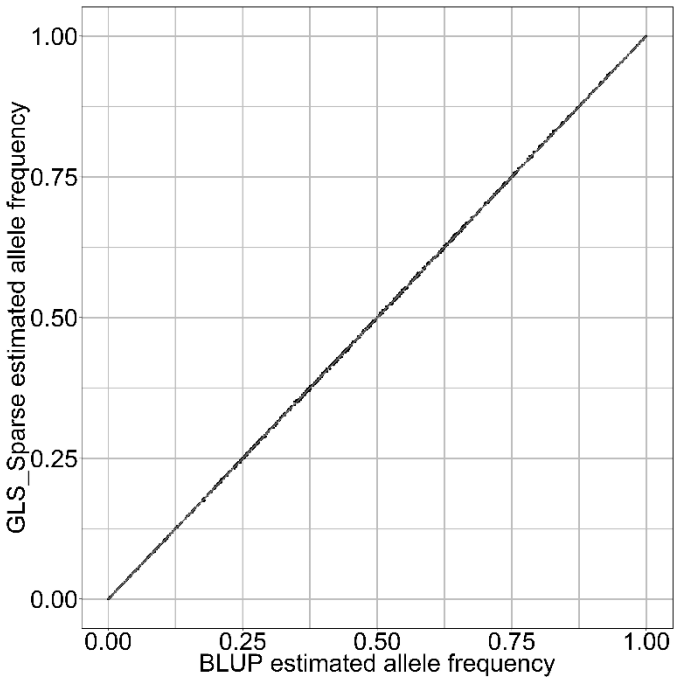
Aldridge Figure 1



Aldridge Figure 2



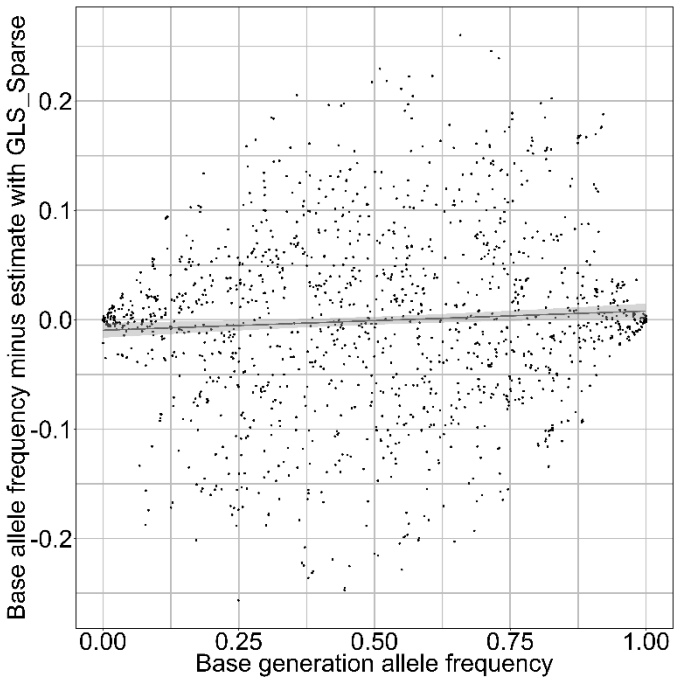
493 Aldridge Figure 3



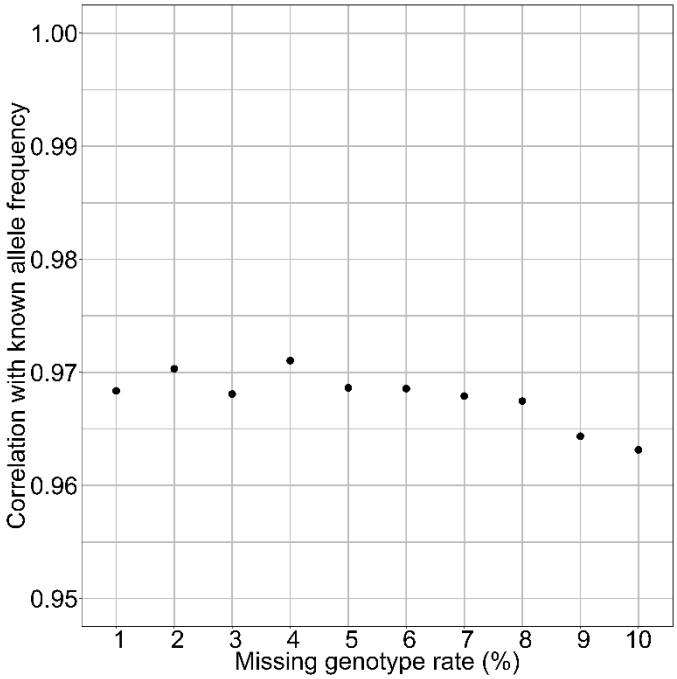
494

495

Aldridge Figure 4



499 Aldridge Figure 5



500

Figure 1: Change in allele frequency between generation 9 and 12 for the base simulation without selection (**top left**) and the base simulation with selection (**top right**). Change in allele frequency between generation 0 and 12 for the base simulation without selection (**bottom left**) and the base simulation with selection (**bottom right**).

Figure 2: Mean time per SNP for MiXBLUP to start and end, solving mixed model equations, with the base simulation dataset.

Figure 3: The allele frequency estimated with BLUP versus GLS_Sparse, for the base simulation with selection.

Figure 4: The relationship between the base generation allele frequency, and the difference between the estimated allele frequency with GLS_Sparse compared to the base generation allele frequency, with a linear regression, for the base simulation with selection.

Figure 5: The relationship between increasing the missing genotype rate and the correlation between estimated frequency with GLS_Sparse and the known base allele frequency.